A. AMENDMENTS TO CLAIMS

Please amend the claims as indicated hereinafter.

1   1.   (PREVIOUSLY PRESENTED) A method of debugging a first software program, the

2   method comprising the steps of:

3   preserving a memory state of a preserved portion of the first software program;

4   dynamically linking a second software program to the first software program without

5   deallocating from volatile memory the first software program;

6   executing the second software program; and

7   if execution of the second software program would otherwise cause modification to

8   targeted data that is in the preserved portion of the first software program,

9   then making a copy of the targeted data and modifying the copy of the

10   targeted data to generate a modified copy of the targeted data without

11   modifying the targeted data that is in the preserved portion of the first

12   software program.

1   2.   (ORIGINAL) The method of Claim 1, further comprising the steps of:

2   publishing in the preserved portion of the first software program a corresponding

3   symbolic name associated with the second software program; and

4   multiple users accessing the second software program is accessed through the

5   corresponding symbolic name.

1   3.   (ORIGINAL) The method of Claim 1, wherein the first software program is a

2   database system.

1 4. (ORIGINAL) The method of Claim 1, wherein the step of preserving a memory

2   state further includes the step of suspending a failed application of the database

3   system.

1 5. (ORIGINAL) The method of Claim 1, further including the step of, in response to

2   a subsequent attempt to access the targeted data in the preserved portion of the

3   first software program, accessing the modified copy of the targeted data.

1 6. (ORIGINAL) The method of Claim 5, wherein the steps of dynamically linking

2   and executing are initiated by a particular user, and wherein the step of accessing

3   the modified copy occurs only if that particular user initiates the subsequent

4   attempt to access the targeted data.

1 7. (ORIGINAL) The method of Claim 1, wherein:

2   the steps of dynamically linking and executing the second software program are

3   performed by a first user;

4   the modified copy is a first modified copy of the targeted data; and

5   the method further comprises the steps of:

6   after the first modified copy has been created for the first user, a second user

7   executing performing an operation which, when executed, would cause

8   modification to the targeted data in the preserved portion; and

9   performing the operation by making a second copy of the targeted data and

10   modifying the second copy to generate a second modified copy of the

11   targeted data, the second modified copy being separate from the first

12   modified copy and from the preserved portion.

1 8. (ORIGINAL) The method of Claim 7, further comprising the steps of:

2   after the first and second modified copies have been created for the first user and

3   second user respectively, a third user dynamically linking and executing a

4   third software program which, when executed, would cause modification to

5   the targeted data in the preserved portion; and

6   making a third copy of the targeted data and modifying the third copy to generate a

7   third modified copy, the third modified copy being separate from the first

8   modified copy, from the second modified copy, and from the preserved

9   portion.

1 9. (PREVIOUSLY PRESENTED) A computer-readable medium bearing instructions

2   for debugging a first software program, the instructions arranged, when executed by

3   one or more processors, to cause the one or more processors to perform the steps of:

4   preserving a memory state of a preserved portion of the first software program;

5   dynamically linking a second software program to the first software program without

6   deallocating from volatile memory the first software program;

7   executing the second software program; and

8   if execution of the second software program would otherwise cause modification to

9   targeted data that is in the preserved portion of the first software program,

10   then making a copy of the targeted data and modifying the copy of the

11   targeted data to generate a modified copy of the targeted data without

12   modifying the targeted data that is in the preserved portion of the first

13   software program.

1   10.   (ORIGINAL) The computer-readable medium of Claim 9, further comprising the

2   steps of:

3   publishing in the preserved portion of the first software program a corresponding

4   symbolic name associated with the second software program; and

5   multiple users accessing the second software program is accessed through the

6   corresponding symbolic name.

1   11.   (ORIGINAL) The computer-readable medium of Claim 9, wherein the first software

2   program is a database system.

1   12.   (ORIGINAL) The computer-readable medium of Claim 9, wherein the step of

2   preserving a memory state further includes the step of suspending a failed

3   application of the database system.

1   13.   (ORIGINAL) The computer-readable medium of Claim 9, further including the

2   step of, in response to a subsequent attempt to access the targeted data in the

3   preserved portion of the first software program, accessing the modified copy of

4   the targeted data.

1   14.   (ORIGINAL) The computer-readable medium of Claim 13, wherein the steps of

2   dynamically linking and executing are initiated by a particular user, and wherein

3   the step of accessing the modified copy occurs only if that particular user initiates

4   the subsequent attempt to access the targeted data.

1   15.   (ORIGINAL) The computer-readable medium of Claim 9, wherein:

2   the steps of dynamically linking and executing the second software program are

3   performed by a first user;

4   the modified copy is a first modified copy of the targeted data; and

5   the method further comprises the steps of:

6   after the first modified copy has been created for the first user, a second user

7   executing performing an operation which, when executed, would cause

8   modification to the targeted data in the preserved portion; and

9   performing the operation by making a second copy of the targeted data and

10   modifying the second copy to generate a second modified copy of the

11   targeted data, the second modified copy being separate from the first

12   modified copy and from the preserved portion.

1   16.   (ORIGINAL) The computer-readable medium of Claim 15, further comprising the

2   steps of:

3 after the first and second modified copies have been created for the first user and

4 second user respectively, a third user dynamically linking and executing a

5 third software program which, when executed, would cause modification to

6 the targeted data in the preserved portion; and

7 making a third copy of the targeted data and modifying the third copy to generate a

8 third modified copy, the third modified copy being separate from the first

9 modified copy, from the second modified copy, and from the preserved

10 portion.

1 17. (PREVIOUSLY PRESENTED) An apparatus for debugging a first software program,

2 wherein the apparatus comprises a memory storing one or more instructions which,

3 when executed by one or more processors, cause the one or more processors to

4 perform the steps of:

5 preserving a memory state of a preserved portion of the first software program;

6 dynamically linking a second software program to the first software program without

7 deallocating from volatile memory the first software program;

8 executing the second software program; and

9 if execution of the second software program would otherwise cause modification to

10 targeted data that is in the preserved portion of the first software program,

11 then making a copy of the targeted data and modifying the copy of the

12 targeted data to generate a modified copy of the targeted data without

13 modifying the targeted data that is in the preserved portion of the first

14 software program.

1   18.   (CURRENTLY AMENDED) The ~~computer-readable medium~~ apparatus of Claim 17,

2   wherein the memory includes one or more additional instructions which, when

3   executed by the one or more processors, cause the one or more processors to perform

4   the additional steps of:

5   publishing in the preserved portion of the first software program a corresponding

6   symbolic name associated with the second software program; and

7   multiple users accessing the second software program is accessed through the

8   corresponding symbolic name.

1   19.   (CURRENTLY AMENDED) The ~~computer-readable medium~~ apparatus of Claim 17,

2   wherein the first software program is a database system.

1   20.   (CURRENTLY AMENDED) The ~~computer-readable medium~~ apparatus of Claim

2   17, wherein the step of preserving a memory state further includes the step of

3   suspending a failed application of the database system.

1   21.   (CURRENTLY AMENDED) The ~~computer-readable medium~~ apparatus of Claim

2   17, wherein the memory includes one or more additional instructions which, when

3   executed by the one or more processors, cause the one or more processors to

4   perform the additional step of, in response to a subsequent attempt to access the

5   targeted data in the preserved portion of the first software program, accessing the

6   modified copy of the targeted data.

1   22.   (CURRENTLY AMENDED) The ~~computer-readable medium~~ <u>apparatus</u> of Claim

2   21, wherein the steps of dynamically linking and executing are initiated by a

3   particular user, and wherein the step of accessing the modified copy occurs only if

4   that particular user initiates the subsequent attempt to access the targeted data.

1   23.   (CURRENTLY AMENDED) The ~~computer-readable medium~~ <u>apparatus</u> of Claim 17,

2   wherein:

3   the steps of dynamically linking and executing the second software program are

4   performed by a first user;

5   the modified copy is a first modified copy of the targeted data; and

6   wherein the memory includes one or more additional instructions which, when

7   executed by the one or more processors, cause the one or more processors to

8   perform the additional steps of:

9   after the first modified copy has been created for the first user, a second user

10   ~~executing~~ <u>performing</u> an operation which, when executed, would cause

11   modification to the targeted data in the preserved portion; and

12   performing the operation by making a second copy of the targeted data and

13   modifying the second copy to generate a second modified copy of the

14   targeted data, the second modified copy being separate from the first

15   modified copy and from the preserved portion.

1   24.   (CURRENTLY AMENDED) The ~~computer-readable medium~~ <u>apparatus</u> of Claim 23,

2   wherein the memory includes one or more additional instructions which, when

3   executed by the one or more processors, cause the one or more processors to perform

4   the additional steps of:

5   after the first and second modified copies have been created for the first user and

6   second user respectively, a third user dynamically linking and executing a

7   third software program which, when executed, would cause modification to

8   the targeted data in the preserved portion; and

9   making a third copy of the targeted data and modifying the third copy to generate a

10   third modified copy, the third modified copy being separate from the first

11   modified copy, from the second modified copy, and from the preserved

12   portion.